

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 01-246657

(43)Date of publication of application : 02.10.1989

(51)Int.Cl.

G06F 15/16

(21)Application number : 63-073089

(71)Applicant : TOSHIBA CORP

(22)Date of filing : 29.03.1988

(72)Inventor : FUJITA JUNICHI

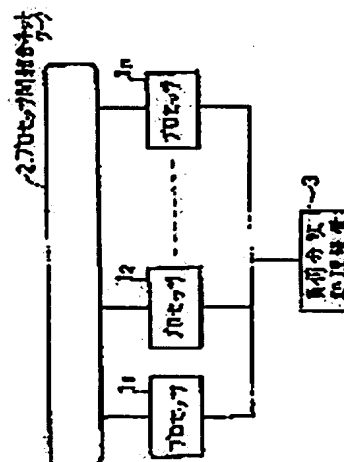
(54) PARALLEL PROCESSING SYSTEM

(57)Abstract:

PURPOSE: To equalize a load and to make efficient the whole by counting an operation rate in the prescribed time of respective processors and allocating a new object to the processor in which a load is minimum.

CONSTITUTION: Plural processors 11W1n are mutually connected through an inter-processor coupling network 2 and connected to a load dispersion processor 3.

Respective processors 11W1n advances the processing by message- communicating respective objects between respective processors. Respective processors 11W1n have a counter to count respective own idle times and transmits the value of the counter to the load dispersion processor 3 for a prescribed time. The load dispersion processor 3 grasps the load condition of respective processors 11W1n based on the value and allocates the new object to a processor with the smallest load.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

⑨ 日本国特許庁(JP)

⑩ 特許出願公開

⑫ 公開特許公報(A) 平1-246657

⑬ Int. Cl.⁴
G 06 F 15/16

識別記号
3 8 0

庁内整理番号
Z-6745-5B

⑭ 公開 平成1年(1989)10月2日

審査請求 未請求 請求項の数 1 (全7頁)

⑮ 発明の名称 並列処理システム

⑯ 特 願 昭63-73089

⑰ 出 願 昭63(1988)3月29日

⑱ 発 明 者 藤 田 純 一 神奈川県川崎市幸区小向東芝町1番地 株式会社東芝総合
研究所内

⑲ 出 願 人 株 式 会 社 東 芝 神奈川県川崎市幸区堀川町72番地

⑳ 代 理 人 弁 理 士 鈴 江 武 彦 外2名

明 細 書

1. 発明の名称

並列処理システム

2. 特許請求の範囲

複数のプロセッサを結合して構成され、各プロセッサに割付けられたオブジェクトを並列に処理する並列処理システムにおいて、

前記各プロセッサの所定時間内における遊休時間若しくは実行時間をカウントし、そのカウント値を上記所定時間毎に通知する手段と、この手段によって通知された上記各カウント値に基づいて前記各プロセッサの負荷状況情報を求め、新たに生成されたオブジェクトを前記負荷状況情報から把握できる最小負荷のプロセッサに割付ける手段とを具備したことを特徴とする並列処理システム。

3. 発明の詳細な説明

〔発明の目的〕

(産業上の利用分野)

本発明は、複数のプロセッサから構成される並列処理システムに係わり、特に各プロセッサの

負荷が分散されるようにオブジェクト(実行単位)の割付けを行う手段を備えた並列処理システムに関する。

(従来の技術)

近年、複数のプロセッサを結合ネットワークを介して結合し、各プロセッサが複数のオブジェクトを並列に実行する並列処理システムの開発が進められている。この並列処理システムでは、各プロセッサの負荷が均等になるようにオブジェクトを割付けることが、処理速度の向上を図る上で重要となる。

従来は並列処理システムの各プロセッサの負荷状況を事前に計算によって求め、各プロセッサへのオブジェクトの割付けはこの計算値に基づいて行うようにしていた。

しかし、並列処理システムの場合、システムの規模が大きくなると、負荷状況を計算によって予測することとが事実上困難になり各プロセッサの負荷が均等になるようなオブジェクトの割付けを静的に行うことは不可能であった。

また、動的にオブジェクトが生成される場合、上記のように静的な割付けでは、並列処理システム全体の負荷分散は考慮されずに任意のプロセッサに割り付けられてしまうという問題があった。

(発明が解決しようとする課題)

このように従来の並列処理システムでは、各プロセッサの負荷が均等になるようにオブジェクトを割付けることができず、処理速度を十分に向上させることができなかった。

本発明はこのような欠点を解消し、並列処理システムの各プロセッサの負荷が均等になるように、オブジェクトをプロセッサに動的に割付けることができ、もって処理速度の向上を図ることが可能な並列処理システムを提供することを目的とする。

[発明の構成]

(課題を解決するための手段)

本発明は、複数のプロセッサが割付けられたオブジェクトを並列に処理する並列処理システムにおいて、前記各プロセッサの所定時間内における遊休時間若しくは実行時間をカウントし、その

カウント値を上記所定時間毎に通知する手段と、この手段によって通知された上記各カウント値に基づいて前記各プロセッサの負荷状況情報を求め、新たに生成されたオブジェクトを前記負荷状況情報から把握できる最小負荷のプロセッサに割付ける手段とを具備したことを特徴としている。

(作用)

本発明によれば、各プロセッサの所定時間内における遊休時間若しくは実行時間をカウントし、そのカウント値を通知する手段を備えているので、各プロセッサの負荷状況を常時監視することができる。そして、本発明では、各プロセッサについてのカウント値に基づいて各プロセッサの負荷状況情報を求め、新たに生成されたオブジェクトを上記負荷状況情報から把握できる最小負荷のプロセッサに割付けるようにしているので、常に各プロセッサの負荷が均等になるようにオブジェクトの割付けを行うことができる。このため、各プロセッサを効率良く稼働させてシステム全体の処理速度を大幅に高めることができる。

(実施例)

以下、図面を参照して本発明の一実施例について説明する。

第1図は本発明の一実施例の並列処理システムの構成図である。複数のプロセッサ 1_1 、 1_2 、 \dots 、 1_n はプロセッサ間結合ネットワーク2を介して互いに接続されるとともに、負荷分散処理装置3にそれぞれ接続されている。各プロセッサ 1_1 、 1_2 、 \dots 、 1_n は、複数のオブジェクトを実行し、各オブジェクトを各プロセッサ間でメッセージ通信することによって、処理を進める。負荷分散処理装置3は、並列処理システムの各プロセッサ $1_1 \sim 1_n$ の負荷が均等になるように各プロセッサ $1_1 \sim 1_n$ にオブジェクトを割り付ける。

第2図はこの並列処理システムを構成する各プロセッサ11の構成図である。

即ち、プロセッサ11は、他のプロセッサからのメッセージを受信するPE間通信部11と、上記メッセージにより起動される複数のオブジェクトを実行/制御する処理装置12と、この処理

装置12の遊休時間(以下「IDLE時間」と呼ぶ)をカウントするIDLEカウンタ21と、このIDLEカウンタ21のカウント動作を制御するカウント制御部22と、所定時間毎にカウント制御部22に割込むための周期タイマー23と、負荷分散処理装置との間の通信を行う負荷分散通信部24とにより構成されている。

又、負荷分散処理装置3は、第3図にその構成を示すように、プロセッサに対するメッセージ通信を行う通信部31と、負荷分散処理装置3の全体の制御を行う負荷分散制御部32と、各プロセッサの負荷状況を記録するプロセッサ負荷記憶部33と、各プロセッサから送信されるオブジェクト・コードを一時記憶するオブジェクト記憶部4と、所定時間毎に負荷分散制御部32に割込むための周期タイマー35とにより構成されている。

次に上記のように構成された並列処理システムの動作について説明する。

まず、第4図のプロセッサの実行例に基づいて、プロセッサの動作を説明する。

処理装置 11 は、P E 間通信部 11 を介してメッセージを受取ると、対応するオブジェクトを開始する。オブジェクトは、一つの実行単位でメッセージを受信することにより起動され、そのメッセージに対する処理を行うと終了する。オブジェクトの終了によって処理装置 11 がメッセージの受信待ち状態となるための receive 命令を実行すると (4 a)、プロセッサ 11 は I D L E (遊休状態) になるので、負荷カウンタ部 13 のカウンタ制御部 22 は I D L E カウンタ 21 のカウンタを開始させる (4 b)。P E 間通信部 11 は他のプロセッサよりメッセージを受信すると (4 c)、その受信メッセージを処理装置 12 に送るとともに、カウンタ制御部 22 にその旨を通知する。これにより、カウンタ制御部 22 は I D L E カウンタのカウンタを停止させる (4 e)。この結果、I D L E カウンタ 21 の値は 50 で停止する。処理装置 12 はその受信メッセージを処理するオブジェクトを実行する。

処理装置 12 はオブジェクトの実行が終了する

と再び receive 命令を実行する (4 f)。負荷カウンタ部 13 のカウンタ制御部 22 は I D L E カウンタ 21 のカウンタを再開させる。そして再び他のプロセッサよりメッセージを受信すると (4 h)、処理装置 12 によるオブジェクトの実行が再開され (4 v)、カウンタ制御部 22 は I D L E カウンタ 21 のカウンタを停止させる (4 j)。この結果、I D L E カウンタ 21 の値は 150 で停止する。

周期タイマー 23 は設定された間隔 (第 4 図の例では I D L E カウンタの基準クロック数で 600) 毎にカウンタ制御部 22 に割り込む (4 k)。カウンタ制御部 22 は、周期タイマー 23 より割り込まれると、I D L E カウンタ 21 の値 (150) を負荷分散通信部 24 を介して負荷分散処理装置 3 に送信し、かつ、I D L E カウンタ 21 の値をクリアする (4 l)。これによって I D L E カウンタ 21 の値は 0 になる。

また、処理装置 11 で新しいオブジェクトが動的に生成される場合、処理装置 11 はその生成さ

れるオブジェクトのコードを負荷分散通信部 24 を介して、負荷分散処理装置 3 に送信し、オブジェクトのプロセッサへの割付けを依頼する。

次に、第 5 図のフローチャート及び第 6 図の実行例に基づいて負荷分散処理装置 3 の動作を説明する。通信部 31 がプロセッサ 11 よりメッセージを受信すると (5 a)、負荷分散制御部 32 は、メッセージを解釈する。もし、I D L E 時間の通知であれば (5 b)、その I D L E 時間より負荷率を次のように計算する (5 d)。

$$\text{付加率} = 100 - (\text{IDLE 時間} / \text{周期タイマーの間隔時間} \times 100)$$

そして、計算した負荷率及び負荷率の平均をプロセッサ負荷記憶部 33 に記録する (5 e)。例えば、プロセッサ負荷記憶部 33 が第 6 図の 6 g の状態で、プロセッサ #3 より I D L E 時間 (150) の通知がメッセージで送信されたとなると (6 a)、負荷分散制御部 32 は、次のように負荷率及び負荷平均計算する。

$$\text{負荷率} = 100 - (150/600 \times 100) = 75$$

$$\text{負荷平均} = (70 + 75) / 2 = 73$$

定められた負荷率及び負荷平均はプロセッサ負荷記憶部 33 のプロセッサ #3 の項目に記録される (6 b)。この結果、プロセッサ負荷記憶部 33 は 6 h の状態になる。

また、プロセッサより新しく生成したオブジェクトの割付け依頼メッセージを受信したならば (5 c)、そのメッセージ内に格納されているオブジェクト・コードを一時的にオブジェクト記憶部 34 に格納する。次に、負荷分散制御部 32 はプロセッサ負荷記憶部 33 を検索し、負荷平均及び最新負荷よりなる負荷状況情報から最も負荷の軽いプロセッサを把握し (5 g)、そのプロセッサに対してオブジェクト記憶部 34 に格納されているオブジェクト・コードを送信し、ロード/実行させる (5 h)。

例えば、プロセッサ負荷記憶部 33 が第 6 図の 6 h の状態で、プロセッサ #0 より新しく生成したオブジェクトの割付け依頼メッセージを受信し

たならば(6c)、負荷分散制御部32はプロセッサ負荷記憶部33を検索し、最も負荷の軽いプロセッサ#1(負荷平均30)に対し、オブジェクト記憶部34に格納されたオブジェクト・コードを送信し、ロード/実行させる(6d)。

また、周期タイマー35から負荷分散制御部32に割込みがかかると(5c)、負荷分散制御部32はプロセッサ負荷記憶部33を検索し、最も負荷の重いプロセッサと最も負荷の軽いプロセッサとを調べ(5d)、その負荷の差がある程度以上あれば、最も負荷の重いプロセッサに対し、そのプロセッサが実行しているオブジェクトのうちの一つを解放し、そのオブジェクト・コードを負荷分散処理装置3に送信するように命令するとともに、そのオブジェクト・コードを消去する。そのオブジェクト・コードを通信部31が受信すると、負荷分散制御部32は受信したオブジェクト・コードを最も負荷の軽いプロセッサに送信し、ロード/実行させる。

例えば、周期タイマーは前述した基準クロック

で60000毎に負荷分散制御部32に割込む(6e)。割込まれた負荷分散制御部32はプロセッサ負荷記憶部33(6fの状態)を検索し、最も負荷の重いプロセッサ#0(負荷平均:100)及び最も負荷の軽いプロセッサ#2(負荷平均:20)を調べる。負荷分散制御部32はプロセッサ#0に対し、実行しているオブジェクトのうちの一つを解放し、そのオブジェクト・コードを負荷分散処理装置3に送信するように命令する。そのオブジェクト・コードを通信部31が受信すると、負荷分散制御部32は受信したコードをプロセッサ#2に送信し、ロード/実行させる。

このように、本システムによれば、並列処理システムを構成する複数のプロセッサの負荷が均等になるように、動的に調整することにより、並列処理システム全体の処理速度を向上させることができる。

なお、上記実施例では、各プロセッサのIDLE時間をカウントしたが、各プロセッサの実行時間をカウントすることによっても負荷状況を把握する

ことは可能である。又、負荷状況情報は、負荷率、負荷平均のみならず、カウント値そのものであっても良い。

【発明の効果】

以上述べたように、本発明によれば並列処理システムを構成する各プロセッサの負荷状況を監視し、動的に生成されるオブジェクトを負荷の軽いプロセッサに割付けることにより、各プロセッサの負荷を常に均等にすることができ、並列処理システム全体の処理速度が向上する。

また、各プロセッサの負荷を逐次ディスプレイ等に表示すれば、並列処理システムの性能評価等に用いることも可能である。

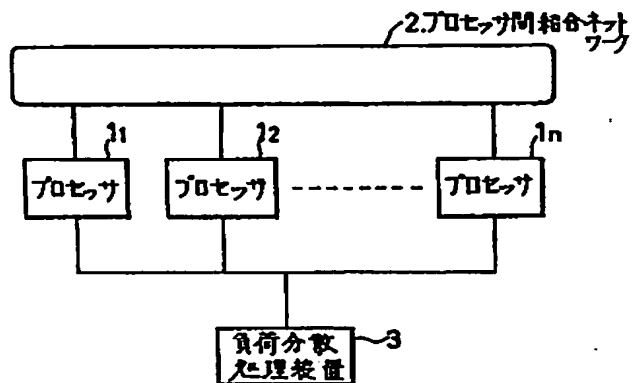
4. 図面の簡単な説明

第1図は本発明の一実施例に係る並列処理システムの全体構成図、第2図は同システムを構成するプロセッサの構成図、第3図は同システムにおける負荷分散処理装置の構成図、第4図は同プロセッサの実行例を示す図、第5図は同負荷分散処理装置の機能を説明するための流れ図、第6図

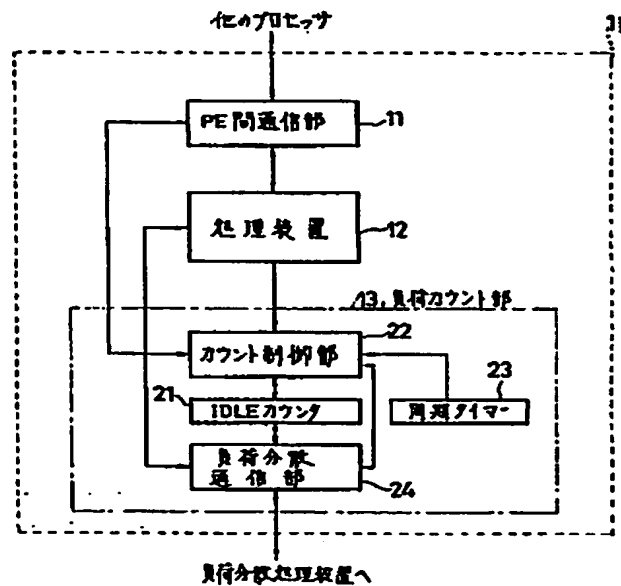
は同負荷分散処理装置の実行例を示す図である。

1₁〜1_n…プロセッサ、2…プロセッサ間結合ネットワーク、3…負荷分散処理装置、11…PE間通信部、12…処理装置、13…負荷カウント部、21…IDLEカウンタ、22…カウント制御部、23…周期タイマー、24…負荷分散通信部、31…通信部、32…負荷分散制御部、32…プロセッサ負荷記憶部、34…オブジェクト記憶部、34…周期タイマー。

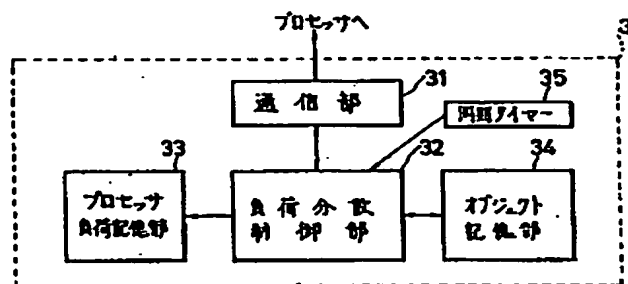
出願人代理人 弁理士 鈴 江 武 彦



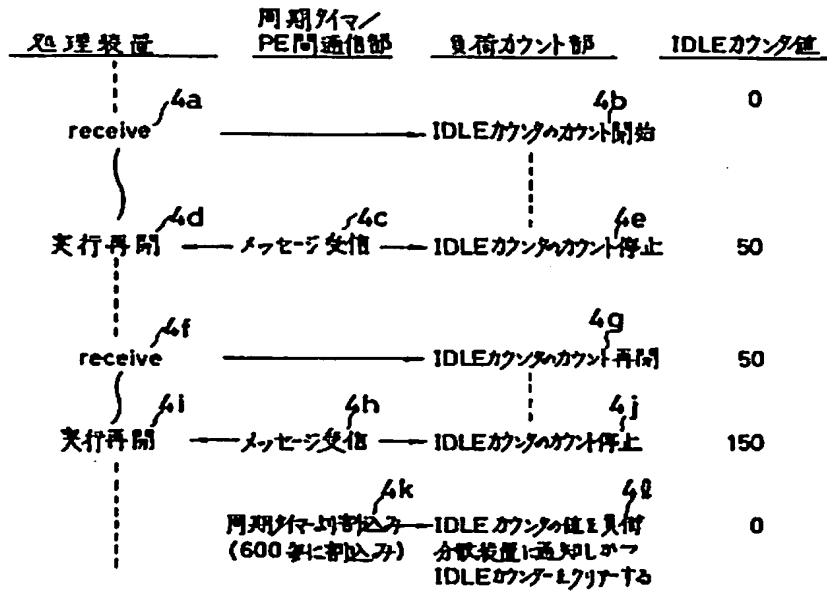
第 1 図



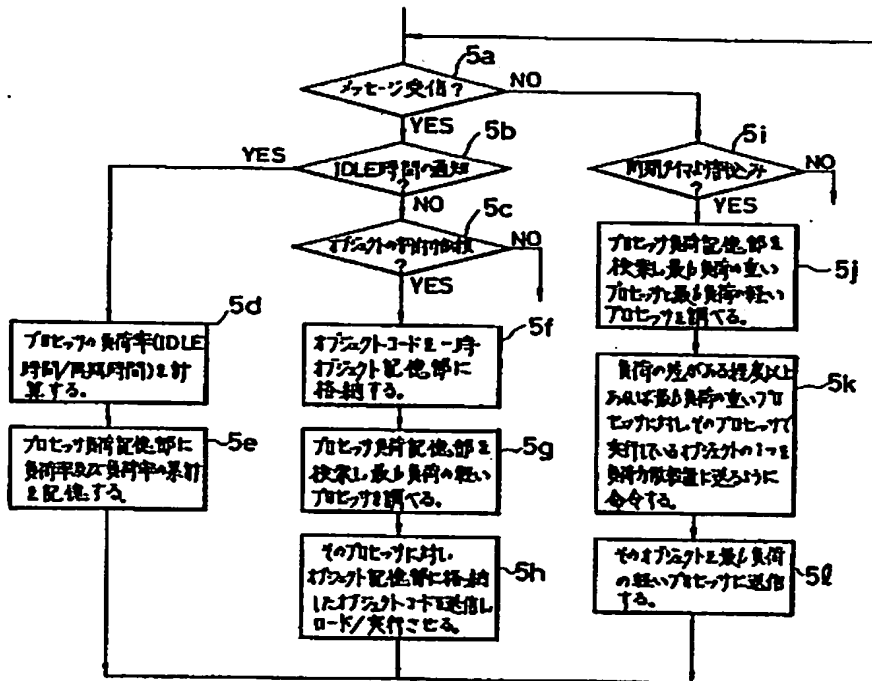
第 2 図



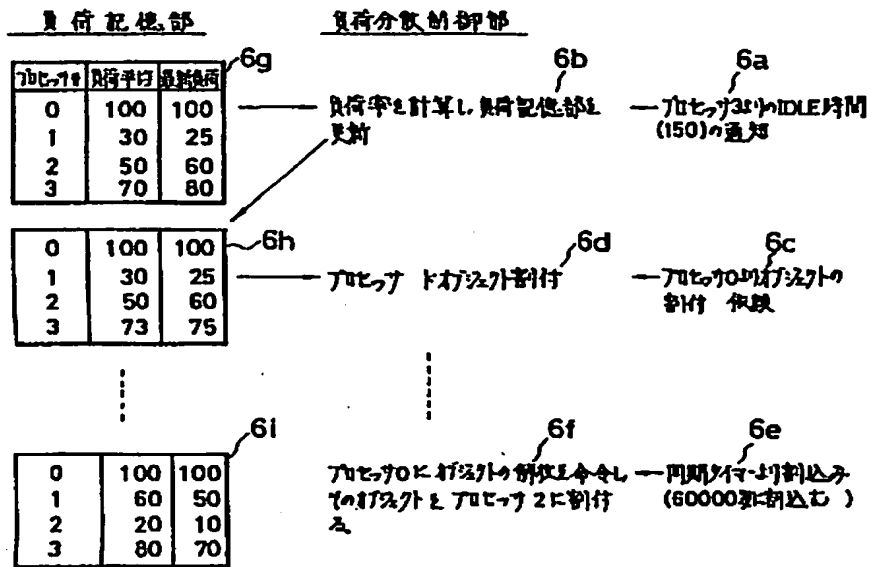
第 3 図



第 4 図



第 5 図



第 6 図